Name: Manu Bhargava Reddy Nannuri

Andrew ID: mnannuri

Homework 3

Statement of Assurance

I certify that all of the material that I submitted is my original work and that it is done only by me.

1 Experiment 1: Baselines

	Ranked Boolean	BM25 BOW	Indri BOW
P@10	0.1500	0.2900	0.2400
P@20	0.1800	0.3050	0.2750
P@30	0.1667	0.3267	0.2967
MAP	0.0566	0.1325	0.1275
Time	00:16	00:17	00:19

2 Experiment 2: Different representations

	Indri BOW (body)	0.2 url 0.2 keywords 0.2 title 0.2 body 0.2 inlink	0.1 url 0.1 keywords 0.2 title 0.5 body 0.1 inlink	0.1 url 0.3 keywords 0.3 title 0.2 body 0.1 inlink	0.3 url 0.1 keywords 0.1 title 0.2 body 0.3 inlink	0.25 url 0.25 keywords 0.25 title 0.00 body 0.25 inlink
P@10	0.2400	0.13	0.2300	0.1800	0.1900	0.2100
P@20	0.2750	0.19	0.2500	0.2050	0.2100	0.2350
P@30	0.2967	0.2033	0.2733	0.2100	0.2267	0.2433
MAP	0.1275	0.1059	0.1174	0.1072	0.1079	0.0937
Time	00:19	00:35	00:27	00:29	00:31	00:31

I organized the fields into 3 categories to measure how important these fields. I could actually pick a field at once but then the number of queries are very small to actually get decisive results.

→ location related like url and inlinks , these are usually characteristic of html pages and their location.

→ Fields like Title and Keywords, which usually summarize the content of the page. These are like the key indicators of a page.

→ Body.

I have varied the weights so that I could see which categories of information are most informative for our information needs and our queries.

I have established a baseline by setting an equal weight for every field. In this case I got the MAP numbers to be 0.1059. As I tend to increase the weightage on different categories that I have made(excluding body), I have seen that the resulting MAP numbers were almost similar with slightly increased Precision. The Body fields is quite important as my experiments have shown. As I assign maximum weightage to the body field, I have seen improved MAP numbers as compared to the cases where I assigned more weightage to the other fields. Clearly for this set of queries, the body field looks to be the most informative flds from my experiments. My results also suggest that not giving the body enough weightage is also pushing down the MAP Numbers from 0.1275 to 0.1174. When I totally ignored the body field, I observed the worst MAP number. So this shows that body might be the most important field for the given query set and the information needs.

Clearly I did not expect the other fields to have such low importance. But this might be due to our choice of query set and using other queries or some other information needs which rely more on titles might lead to better results. Our currently query set has very few queries which have terms that could all be present in the title fields or keywords field(for eg AVP is one case. and this might most likely be present in inlinks or urls.) Majority of them are common words and mostly will be definitely present in the body in whole(all of them would be present in the body).

I have also observed an increase in the running time for queries using multiple representations. This is due an increase in the number of query terms when se use multiple representations. In the bow case with only words we are computing for the number of terms in the query. whereas in the multiple representations case, this is 5X times the number of query terms as in the original base case.

3 Experiment 3: Sequential dependency models

Example Query: Provide your structured query for query “fickle creek farm”.

Ans: *102:#WAND(0.2 #AND(fickle creek farm) 0.2 #AND(#NEAR/1(creek farm) #NEAR/1(fickle creek)) 0.6 #AND(#WINDOW/8(creek farm) #WINDOW/8(fickle creek)))*

	Indri BOW (body)	**0.00 AND 1 NEAR 0 WINDOW**	**0.00 AND 0.00 NEAR 1 WINDOW**	**0.33 AND 0.33 NEAR 0.33 WINDOW**	**0.00 AND 0.5 NEAR 0.5 WINDOW**	**0.2 AND 0.6 NEAR 0.2 WINDOW**	**0.2 AND 0.2 NEAR 0.6 WINDOW**
P@10	0.2400	0.3700	0.2900	0.3700	0.3700	0.3600	0.3700
P@20	0.2750	0.3750	0.3400	0.3650	0.3700	0.3700	0.3700
P@30	0.2967	0.3733	0.3500	0.3667	0.3800	0.3733	0.3800
MAP	0.1275	0.1769	0.1540	0.1934	0.1914	0.1819	0.1840
Time	00:19*	0:03	0:04	0:25	0:05	0:27	0:27

* This measurement was made on a server for the last experiment and now I have got a new laptop. On my current machine this exact number is 12.45seconds.

Describe how you set the weights for the different components of the sequential dependency model.

Since there were few weights to set. I tried to measure the importance of each component of the sequential dependency model. Then I tried measuring the performance when all of them got the similar weight. Then I tried keeping one weight high and the other 2 low.

Discuss any trends that you observe; whether the more complex query behaved as you expected; whether the improvement in accuracy (if any) is worth the increased computational cost; and any other observations that you may have.

The sequential dependence model definitely improved the performance of our search engines.

I measured the importance of each of the components of the SDM and found that the second component i.e the ngram matches. was contributing more than the others. But both the short window matches and the ngram matches components are better than the bag of words as the map numbers in the respective cases were 0.1768 and 0.1540 compared to 0.1275 in the bag of word matches only. I have observed a better map score when I took only a combination of the ngram matches and short window matches. But the best performance measure was observed when I assigned all of them equal weights. This shows us that the bag of words matches are also important and cannot be ignored completely as there might be few cases where in the query might not actually involve phrases. In such, cases the bag of words model would contribute the most.

These numbers could be explained by our specific choice of our query set. The queries that we are using in our testing set have lots of phrases like living in india, brooks brothers sale, fickle creek farm, cheap internet, lower heart rate etc. So our query set is actually engineered to perform well with proximity operators like near and windows rather than simple bag of words matches. Hence, when we raise the

weights for these proximity operators we get improved results. Since we also have some queries which are not totally reliant on phrases, the bag of words model is also required to have good results.

I have found similar running times for both the multiple representations queries and the queries which formed using the sequential dependence model. There is a significant improvement in the MAP numbers in the SDM compared to the multiple representation model. This shows that SDM can be preferred over multiple representation model.

4 Experiment 4: Multiple representations + SDMs

Example Query: Provide your structured query for query "fickle creek farm".

102:#WAND(w1 #AND (#WSUM (0.1 fickle.url 0.1 fickle.keywords 0.1 fickle.title 0.6 fickle.body 0.1 fickle.inlink) #WSUM (0.1 creek.url 0.1 creek.keywords 0.1 creek.title 0.6 creek.body 0.1 creek.inlink) #WSUM (0.1 farm.url 0.1 farm.keywords 0.1 farm.title 0.6 farm.body 0.1 farm.inlink)) w2 #WAND(0.5 #AND(#NEAR/1(creek farm) #NEAR/1(fickle creek)) 0.5 #AND(#WINDOW/8(creek farm) #WINDOW/8(fickle creek))))

	Indri BOW (body)	**w=1.0 (Exp 2)**	**0.85**	**0.70**	**0.50**	**0.35**	**0.15**	**w=0.0 (Exp 3)**
P@10	0.2400	0.2500	0.3100	0.3500	0.3800	0.4000	0.4000	0.3700
P@20	0.2750	0.2500	0.3150	0.3400	0.3650	0.3850	0.3900	0.3700
P@30	0.2967	0.2733	0.3233	0.3467	0.3667	0.3700	0.3667	0.3800
MAP	0.1275	0.1194	0.1480	0.1624	0.1842	0.1933	0.1930	0.1914
Time	00:19*	00:15	00:18	00:20	00:20	00:19	00:20	0:05

***** This measurement was made on a server for the last experiment and now I have got a new laptop. On my current machine this exact number is 12.45seconds.

Discuss any trends that you observe; whether the more complex query behaved as you expected; whether the improvement in accuracy (if any) is worth the increased computational cost; and any other observations that you may have.

My w1 weight corresponds to the multiple representation which has maximum weights on body and very less on the others. The (1-w1) weight corresponded to a SDM query which had no weights on the body fields.

I observed that as w1 goes from 1 to 0, the map values tend to rise consistently and then attain a peak around 0.35 and then again falls as it goes to 1.

The query set behaved as I expected it to behave. I expected it to peak around a value when ⅔ of the weights go to the SDM part and ⅓ goes to the multiple representations part. This is because my SDM part had no weights assigned to the bag of words methods and this got aptly fulfilled by the multiple

representation part which relies only on the bag of words part. Hence yes, I got the results that I expected. My hunch is that even if I picked a sdm with a weights split of(0.3,0.3,0.3), then the peak would come when w1=0. This might be true because that is the ideal split as we have found from SDM and adding more weights to the bag of queries matches across multiple representations would not improve results as we have seen in experiment 2 that the BOW only from the body field received maximum MAP score.

The increase in time is actually not so much compared to the best SDM query which had a similar MAP value and a less running time. The increased running time was because the 1st part of this query had to query more fields than just the body in the SDM case. Moreover, our queries are heavily reliant on the body field, hence it would be more appropriate to take a diverse query set and then verify if the increased complexity is a good payoff for the increased running time.